PATENT
ATTY. DOCKET NO. IBM/286

Applicant	:	Bestgen et al.	Art Unit: 2162
Serial No.	:	10/758,485	Examiner: Giovanna B. Colan
Filed	:	January 15, 2004	
For	:	LOOKAHEAD PREDICATE GENERATION FOR JOIN COSTING AND OPTIMIZATION	

Mail Stop Amendment
Commissioner of Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Via Facsimile

RESPONSE TO NOTICE OF NON-COMPLIANT APPEAL BRIEF

This paper and the attached Appeal Brief is in response to the Notice of Non-Compliant Appeal Brief mailed August 6, 2007.

If any petition for extension of time is necessary to accompany this communication, please consider this paper a petition for such an extension of time, and apply the appropriate extension of time fee to Deposit Account 23-3000. If any other charges or credits are necessary to complete this communication, please apply them to Deposit Account 23-3000.

Respectfully submitted,

/ Thomas W. Humphrey /

Thomas W. Humphrey
Reg. No. 34,353

Wood, Herron & Evans, L.L.P.
2700 Carew Tower
441 Vine Street
Cincinnati, OH 45202-2917
Voice: (513) 241-2324/Facsimile: (513) 241-6234

Certificate of Mailing or Facsimile

I hereby certify that this correspondence and any enclosures are being deposited with the United States Postal Service via U.S. Mail, Prepaid First Class Postage or transmitted via facsimile to the number as provided by the United States Patent and Trademark Office on the date indicated below.

/ Thomas W. Humphrey / January 14, 2008
Thomas W. Humphrey Date
Reg. No. 34,353

PATENT
ATTY. DOCKET NO. IBM/286

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

Ex parte Robert Joseph Bestgen et al.

Appeal No. _____

Serial No.:	**10/758,485**
Filed:	**January 15, 2004**
Group Art Unit:	**2162**
Examiner:	**Giovanna B. Colan**
Applicant:	**Robert Joseph Bestgen et al.**
Title:	**LOOKAHEAD PREDICATE GENERATION FOR JOIN COSTING AND OPTIMIZATION**

Cincinnati, Ohio 45202

Original Submission September 24, 2007
Resubmitted January 14, 2008
Via Facsimile

APPEAL BRIEF

This brief is in furtherance of Applicant's Notice of Appeal filed July 24, 2007, appealing the decision of the Examiner dated January 24, 2007 finally rejecting claims 1-19. A copy of the claims appears in the Appendix to this brief.

Certificate of Mailing or Facsimile Transmission

I hereby certify that this correspondence and any enclosures are being deposited with the United States Postal Service via U.S. Mail, Prepaid First Class Postage or transmitted via facsimile to the number as provided by the United States Patent and Trademark Office on the date indicated below.

/Thomas W. Humphrey/ January 14, 2008
Thomas W. Humphrey Date
Reg. No. 34,353

Real Party In Interest

The real party in interest in this appeal is INTERNATIONAL BUSINESS MACHINES CORPORATION, a(n) corporation of New York having a place of business at New Orchard Road, Armonk, New York 10504.

Related Appeals and Interferences

There are no such appeals or interferences.

Status of Claims

Total Number of Claims in the Application

Claims in the application are: 19

Status of all the Claims

1. Claims cancelled: NONE
2. Claims withdrawn from consideration but not cancelled: NONE
3. Claims objected to: 1-19
4. Claims allowed or confirmed: NONE
5. Claims rejected: 1-19

Claims on Appeal

The claims are appeal are Claims 1-19. Claims 1, 9 and 17 were amended in the Response to Office Action filed October 13, 2006.

Status of Amendments

There are no amendments pending.

Summary of Claimed Subject Matter as to Independent Claim 1

Claim 1 is a method claim. An example of the subject matter of independent Claim 1 is described in the specification on page 11 line 1 through page 13 line 5, and Fig. 4, reference numbers 200-230 of the drawings.

The present application, as explained at the noted location, and in the background and specification, is directed to evaluating join predicates in a query to determine selectivity of a join between a first and second relation. An example of the issue is provided in the Background, pages 2-3, which describes a query seeking total sales figures for a product in the New England region. This query involves the join of a (relatively small) SKU table, (relatively large) Sales table, and (relatively small) Stores table. Furthermore, since the search is limited to New England, the Sales table join will be very selective -- potentially involving only one record for a single New England store. The background of the present application explains methods in which such a query might be handled inefficiently.

An objective the present invention is set forth on page 4, specifically, to provide a "method for identifying when a query may be more efficiently implemented with nonconventional join techniques in more diverse circumstances than has previously been the case." The application proceeds to describe methods for evaluating a join of two relations found in a query, to determine whether that join will be selective; so that the join may be performed in an efficient manner "by the prior application of a look up predicate based upon the second relation in the join."

The use of lookup predicates is described in the Background, page 3 -- a lookup predicate is an intermediate result of all tuples matching a selection criterion on an outer table

of a join, and when a join is performed with a lookup predicate, the join involves comparing all tuples in the selection criterion to the tuples in the inner table at the same time, streamlining processing as compared to conventional methods. See, for example, steps 206-214 of Fig. 4, in which an estimate is made of the selectivity of a join when a selection criterion is applied to one table of the join, to determine if the join is beneficially reductive.

The present claims, therefore, are directed to "evaluating join predicates in [a] query to determine whether a join involving a first relation and a second relation will be reductive of [the] first relation," as is shown in steps 206-214 of Fig. 4 and discussed at page 11, line 12 through page 12, line 2, and "identifying a join involving [the] first and second relations that will be reductive of [the] first relation", as shown in step 214-216 and discussed at page 12 lines 3-17, and then "performing [the] query by the prior application of a look-ahead predicate," as described.

Summary of Claimed Subject Matter as to Independent Claim 9

Independent claim 9 is an apparatus claim. An example of the subject matter of independent Claim 9 is also described in the specification on page 11 line 1 through page 13 line 5, and Fig. 4, reference numbers 200-230 of the drawings, as discussed above. That subject matter is summarized above. Claim 9, like claim 1, is directed to "evaluating join predicates in [a] query to determine whether a join involving a first relation and a second relation will be reductive of [the] first relation," as is shown in steps 206-214 of Fig. 4 and discussed at page 11, line 12 through page 12, line 2, and "identifying a join involving [the] first and second relations that will be reductive of [the] first relation", as shown in step 214-216 and discussed at page 12 lines 3-17, and then "performing [the] query by the prior application of a look-ahead predicate," as described.

Summary of Claimed Subject Matter as to Independent Claim 17

Independent claim 17 is a program product claim. An example of the subject matter of independent Claim 17 is also described in the specification on page 11 line 1 through page 13 line 5, and Fig. 4, reference numbers 200-230 of the drawings, as discussed above. That subject matter is summarized above. Claim 17, like claims 1 and 9, is directed to a program product with a relational database system adapted to perform a query by "evaluating join predicates in [a] query to determine whether a join involving a first relation and a second relation will be reductive of [the] first relation," as is shown in steps 206-214 of Fig. 4 and discussed at page 11, line 12 through page 12, line 2, and "identifying a join involving [the] first and second relations that will be reductive of [the] first relation", as shown in step 214-216 and discussed at page 12 lines 3-17, and then "performing [the] query by the prior application of a look-ahead predicate," as described.

Grounds of Rejection

Whether any of claims 1, 9 and 17 are indefinite under 35 USC 112.

Whether the subject matter of any of claims 1-19 is anticipated by Agarawal, U.S. Patent 6,513,029 (Agarawal).

RECEIVED
CENTRAL FAX CENTER
JAN 14 2008

Argument - Rejections under 35 U.S.C. § 112

The Examiner's rejection of claim 1, 9 and 17 is based upon the language in those claims: "performing said query by the prior application of a look-ahead predicate based upon the second relation in the join". The Examiner asserts that "the prior application" in this clause, lacks antecedent basis. However, the Examiner is misreading the claim language; "the prior application" is not referring to any other language in the claim, but rather it is a description of the manner of performance of the query – the query is performed by prior, i.e., initial, application of a predicate that is based on the second relation in the join. This is what is referenced in the summary of the invention, page 5, lines 16-21: "the relational database system evaluates the computational expense of generating a look-ahead predicate comprising the tuples of the second relation that match any selection criterion, and this expense is compared to the computational savings that result from the join reduction. In the event the formation of the look-ahead predicate is beneficial, processing of the query forms and utilizes the look-ahead in the join of the first and second relations." Applicant submits that the language in claims 1, 9 and 17 referencing "the prior application of a look-ahead predicate" is clear and not indefinite, and has not amended it for that reason. However, Applicant is open to amending this language, e..g, by the deletion of "the" before "prior", if the Examiner views a particular amendment as beneficial in overcoming this objection.

Argument - Rejections under 35 U.S.C. § 102

The Examiner has rejected all claims as anticipated by Agarawal. However, Agarawal nowhere deals with the issues discussed, specifically, Agarawal does not relate to "evaluating

join predicates in [a] query to determine whether a join involving a first relation and a second relation will be reductive of [the] first relation", or to "performing a query by the prior application of a look-ahead predicate based upon the second relation in the join," as claimed herein.

In contrast to these issues and concept, Agarawal is directed to a type of index known as a "materialized view". Agrawal explains that a "materialized view" "is some view of the data, such as the results of a query, which have been materialized. A materialized view may not be directly tied to a query. A materialized view has some characteristics that a traditional index does not have. Materialized views may be defined over multiple tables, and can have selections and group-by over multiple columns. It can have selection and grouping of columns." Agarawal explains that query performance can be improved using "materialized views", specifically by shortening the query by using the "materialized view" to form intermediate query results, instead of generating those intermediate results from scratch from the relations of the database.

Firstly, Applicant notes that the Agarawal text cited by the Examiner is directed specifically to the determination of which materialized views should be formed and retained in memory, i.e., identifying those materialized views that will be beneficial and not consume excessive memory. The Examiner's references to Fig.6 and to the text in Fig. 15 all relate to this main intention of Agarawal to find efficient and useful "materialized views".

While Agarawal is thus related generally to database processing, it bears no relation to the claimed invention. First, the selection of desirable "materialized views" is not query processing, but is a form of indexing: a precursor to performing a query. The Examiner's

Final Rejection seems to have missed this point. The "materialized view" is an index for use in later queries, and the discussion in Agrawal regarding forming "materialized views", and deciding which should be formed, is not about answering a particular query, but rather is directed to generally finding the best views for future, unknown queries.

As a consequence, there is nothing in Agarawal that relates to "performing a query" by "evaluating join predicates in [the] query", since at the time that Agarawal is identifying "materialized views" there is no query, only the hypothetical possibility of a future query for which a "materialized view" may be useful. The claim language, directed to actually "performing a query" and steps taken during performance of the query, thus is not and could not be anticipated by language regarding creation of an index or "materialized views" when there is not yet a query to be processed.

Second, Applicant notes that the creation of "materialized views" does not involve the claim steps, most particularly the claim step of determining whether a join will be reductive of a relation so that the query can be performed "by the prior application of a look-ahead predicate". There is simply no discussion in the Agarawal text cited by the Examiner of the use of a look-ahead predicate or the determination of when one should be used.

The Examiner cites to columns 15 and 16 of Agarawal regarding the decision of whether to store a materialized view or the "parents" of that view. As stated by Agarawal at col. 15 lines 44 et seq: "The goal of 603 and 703 in the MergeViewPair algorithms is to eliminate merged materialized views from being generated that are likely to be 'much less useful' than both its parents." This text goes on to explain specifically how the size of a merged materialized view should be evaluated to decide whether the merged view is

sufficiently "useful". While this text is interesting, it does not relate to the evaluation of a particular query, nor does it relate to creating a "look ahead predicate" based upon a relation of a join in a query, nor does it relate to evaluating whether a join is "reductive of [the] first relation of the join", as claimed. The Examiner has nowhere shown a decision process in query processing which determines whether a join is "reductive" and if so, applies a "look ahead predicate" based upon a relation in the join.

Furthermore, although the Examiner has identified mentions of selection criteria in Fig. 6 and at col. 16, lines 49-55, these mentions do not refer to the evaluation of a particular query having a particular selection criterion therein. Claims 2 and 10, however, each refer to a second relation that is subject to a selection criterion, and recite "evaluating whether that selection criterion effects a join rejection"; This subject matter cannot be found in Agarawal. For this reason, Applicant submits that claims 2 and 10 are independently patentable from claims 1 and 9.

Accordingly, Applicant submits that the Examiner's rejection is in error and a reversal of the rejection and allowance of the claims is therefore requested.

Respectfully submitted,
Wood, Herron & Evans, L.L.P.

By _/Thomas W. Humphrey /___
Thomas W. Humphrey
Reg. No. 34,353

2700 Carew Tower
441 Vine Street
Cincinnati, OH 45202-2917
Voice: (513) 241-2324
Facsimile: (513) 241-6234

RECEIVED
CENTRAL FAX CENTER
JAN 14 2008

Claim Appendix

1. (previously presented) A method of performing a query in a relational database system by operating upon a plurality of relations each comprising a plurality of tuples formed over a plurality of attributes, comprising:

evaluating join predicates in said query to determine whether a join involving a first relation and a second relation will be reductive of said first relation, identifying a join involving said first and second relations that will be reductive of said first relation, and performing said query by the prior application of a look-ahead predicate based upon the second relation in the join.

2. (original) The method of claim 1 further comprising determining whether said second relation involved in the join is subject to a selection criterion, and evaluating whether that selection criterion effects a join reduction.

3. (original) The method of claim 2 wherein an amount of join reduction effected by a selection criterion is determined by identifying whether the number of rows in the join result will be smaller than the number of rows in the first relation.

4. (original) The method of claim 2 wherein, upon identifying a join reduction involving a first and a second relation, and a selection criterion on the second relation, the potential benefit of that join reduction is assessed.

5. (original) The method of claim 4 further comprising evaluating the computational expense of generating a look-ahead predicate comprising the tuples of the second relation matching the selection criterion, and comparing said expense to computational savings that result from the join reduction.

6. (original) The method of claim 5 further comprising, upon identifying a beneficial look-ahead predicate, processing the query by forming and utilizing the look-ahead predicate as a selection criterion on the second relations.

7. (original) The method of claim 5 further comprising identifying the most beneficial look-ahead predicate among all potential joins of relations in said query, through iterative analysis of all possible joins.

8. (original) The method of claim 7 further comprising iteratively analyzing all possible joins of the remaining relations and the look-ahead predicate to locate further beneficial look-ahead predicates.

9. (previously presented) Apparatus for performing a query in a relational database system by operating upon a plurality of relations each comprising a plurality of tuples formed over a plurality of attributes, comprising:

a data storage device storing said relations, and

a processor evaluating join predicates in said query to determine whether a join involving a first relation and a second relation will be reductive of said

first relation, identifying a join involving said first and second relations that will be reductive of said first relation, and performing said query by the prior application of a look-ahead predicate based upon the second relation in the join.

10. (original) The apparatus of claim 9 wherein said processor determines whether a relation involved in the join is subject to a selection criterion, and evaluates whether that selection criterion effects a join reduction.

11. (original) The apparatus of claim 10 wherein an amount of join reduction effected by a selection criterion is determined by identifying whether the number of rows in the join result will be smaller than the number of rows in the first relation.

12. (original) The apparatus of claim 10 wherein, upon identifying a join reduction involving a first and a second relation, and a selection criterion on the second relation, said processor assesses the potential benefit of that join reduction.

13. (original) The apparatus of claim 12 wherein said processor evaluates the computational expense of generating a look-ahead predicate comprising the tuples of the second relation matching the selection criterion, and comparing said expense to computational savings that result from the join reduction.

14. (original) The apparatus of claim 13 wherein, upon identifying a beneficial look-ahead predicate, said processor processes the query by forming and utilizing the look-ahead predicate as a selection criterion on the second relations.

15. (original) The apparatus of claim 13 wherein said processor identifies the most beneficial look-ahead predicate among all potential joins of relations in said query, through iterative analysis of all possible joins.

16. (original) The apparatus of claim 15 wherein said processor iteratively analyzes all possible joins of the remaining relations and the look-ahead predicate to locate further beneficial look-ahead predicates.

17. (previously presented) A program product comprising:

a relational database comprising one or more relations, each relation comprising one or more tuples on one or more attributes, and

relational database system adapted to perform a query on said relational database by evaluating join predicates in said query to determine whether a join involving a first relation and a second relation will be reductive of said first relation, identify a join involving said first and second relations that will be reductive of said first relation, and perform said query by the prior application of a look-ahead predicate based upon the second relation in the join, and

signal bearing media bearing the relational database and the relational database system.

18. (original) The program product of claim 17 wherein the signal bearing media comprises transmission media.

19. (original) The program product of claim 17 wherein the signal bearing media comprises recordable media.

Evidence Appendix

None.

Related Proceedings Appendix

None.

TABLE OF CONTENTS